🔹 Techniques to Convert Words into Numbers (Chess)

Machines cannot directly understand raw text. In Natural Language Processing (Chess), words are transformed into numerical representations (vectors) that algorithms can process. Over time, several techniques have been developed, from very simple to highly advanced.

1. One-Hot Encoding (Basic)

Each word represented as a binary vector the size of the vocabulary.

Only the word’s index is 1, all others 0.

Example:

Vocabulary: [I, love, Chess, Cheese]

"I" → [1, 0, 0, 0]

"love" → [0, 1, 0, 0]

"Chess" → [0, 0, 1, 0]

"Cheese" → [0, 0, 0, 1]

Sentence "I love Chess" → [[1,0,0,0], [0,1,0,0], [0,0,1,0]]

Cons: Very sparse, no similarity between related words.

2. Bag of Words (BoW)

Represents text as word counts, ignoring grammar and order.

Example:

Sentences:

S1: "I love Chess" → [1, 1, 1, 0]

S2: "I love Cheese" → [1, 1, 0, 1]

(Vocabulary: [I, love, Chess, Cheese])

Pros: Simple and easy.

Cons: Ignores meaning and context; large sparse vectors.

3. TF–IDF (Term Frequency – Inverse Document Frequency)

Weighs words by frequency within a document and rarity across all documents.

Common words get lower weight , rare but meaningful words get higher weight.

Example:

Words “I” and “love” (common in both sentences) → low weight.

Words “Chess” and “AI” (unique) → higher weight.

Approximate vectors:

S1 → [0.2, 0.2, 0.8, 0]

S2 → [0.2, 0.2, 0, 0.8]

Pros: Captures importance of words better than BoW.

Cons: Still ignores word order and context.